arXiv:1903.02460v2 [cs.CR] 6 Jul 2019

A Survey of Network-based Intrusion Detection Data Sets

Markus Ring, Sarah Wunderlich, Deniz Scheuring, Dieter Landes and Andreas Hotho

***Abstract*—Labeled data sets are necessary to train and evaluate anomaly-based network intrusion detection systems. This work provides a focused literature survey of data sets for network-based intrusion detection and describes the underlying packet- and flow-based network data in detail. The paper identifies 15 different properties to assess the suitability of individual data sets for specific evaluation scenarios. These properties cover a wide range of criteria and are grouped into five categories such as data volume or recording environment for offering a structured search. Based on these properties, a comprehensive overview of existing data sets is given. This overview also highlights the peculiarities of each data set. Furthermore, this work briefly touches upon other sources for network-based data such as traffic generators and data repositories. Finally, we discuss our observations and provide some recommendations for the use and the creation of network-based data sets.**

***Index Terms*—Intrusion Detection, IDS, NIDS, Data Sets, Evaluation, Data Mining**

I. Introduction

IT security is an important issue and much effort has been spent in the research of intrusion and insider threat detection. Many contributions have been published for processing security-related data [1]–[4], detecting botnets [5]–[8], port scans [9]–[12], brute force attacks [13]–[16] and so on. All these works have in common that they require representative network-based data sets. Furthermore, benchmark data sets are a good basis to evaluate and compare the quality of different network intrusion detection systems (NIDS). Given a labeled data set in which each data point is assigned to the class normal or attack, the number of detected attacks or the number of false alarms may be used as evaluation criteria. Unfortunately, there are not too many representative data sets around. According to Sommer and Paxson [17] (2010), the lack of representative publicly available data sets constitutes one of the biggest challenges for anomaly-based intrusion detection. Similar statements are made by Malowidzki et al. [18] (2015) and Haider et al. [19] (2017). However, the community is working on this problem as several intrusion detection data sets have been published over the last years. In particular, the Australian Centre for Cyber Security published the UNSW-NB15 [20] data set, the University of Coburg published the CIDDS-001 [21] data set, or the University of New Brunswick published the CICIDS 2017 [22] data set. More data sets can be expected in the future. However, there is no overall index of existing data sets and it is hard to keep track of the latest developments.

This work provides a literature survey of existing network-based intrusion detection data sets. At first, the underlying data are investigated in more detail. Network-based data appear in packet-based or flow-based format. While flow-based data contain only meta information about network connections, packet-based data also contain payload. Then, this paper analyzes and groups different data set properties which are often used in literature to evaluate the quality of network-based data sets. The main contribution of this survey is an exhaustive literature overview of network-based data sets and an analysis as to which data set fulfills which data set properties. The paper focuses on attack scenarios within data sets and highlights relations between the data sets. Furthermore, we briefly touch upon traffic generators and data repositories as further sources for network traffic besides typical data sets and provide some observations and recommendations. As a primary benefit, this survey establishes a collection of data set properties as a basis for comparing available data sets and for identifying suitable data sets, given specific evaluation scenarios. Further, we created a website [1] which references to all mentioned data sets and data repositories and we intend to update this website.

The rest of the paper is organized as follows. The next section discusses related work. Section III analyzes packet- and flow-based network data in more detail. Section IV discusses typical data set properties which are often used in the literature to evaluate the quality of intrusion detection data sets. Section V gives an overview of existing data sets and checks each data set against the identified properties of Section IV. Section VI briefly touches upon further sources for network-based data. Observations and recommendations are discussed in Section VII before the paper concludes with a summary.

II. Related Work

This section reviews related work on network-based data sets for intrusion detection. It should be noted that host-based intrusion detection data sets like ADFA [23] are not considered in this paper. Interested readers may find details on host-based intrusion detection data in Glass-Vanderlan et al. [24].

Malowidzki et al. [18] discuss missing data sets as a significant problem for intrusion detection, set up requirements for good data sets, and list available data sets. Koch et al. [25]

Markus Ring, Sarah Wunderlich, Deniz Scheuring and Dieter Landes were with the Department of Electrical Engineering and Computer Science, Coburg University of Applied Sciences, 96450 Coburg, Germany (e-mail: markus.ring@hs-coburg.de, sarah.wunderlich@hs-coburg.de, deniz.brix@stud.hs-coburg.de, dieter.landes@hs-coburg.de)

Andreas Hotho was with Data Mining and Information Retrieval Group, University of Würzburg, 97074 Würzburg, Germany (e-mail: hotho@informatik.uni-wuerzburg.de)

[1] http://www.dmir.uni-wuerzburg.de/datasets/nids-ds

provide another overview of intrusion detection data sets, analyze 13 data sources, and evaluate them with respect to 8 data set properties. Nehinbe [26] provides a critical evaluation of data sets for IDS and intrusion prevention systems (IPS). The author examines seven data sets from different sources (e.g. DARPA data sets and DEFCON data sets), highlights their limitations, and suggests methods for creating more realistic data sets. Since many data sets are published in the last four years, we continue previous work [18], [25], [26] from 2011 to 2015, but offer a more up-to-date and more detailed overview than our predecessors.

While many data set papers (e.g., CIDDS-002 [27], ISCX [28] or UGR'16 [29]) give just a brief overview of some intrusion detection data sets, Sharafaldin et al. [30] provide a more exhaustive review. Their main contribution is a new framework for generating intrusion detection data sets. Sharafaldin et al. also analyze 11 available intrusion detection data sets and evaluate them with respect to 11 data set properties. In contrast to earlier data set papers, our work focuses on providing a neutral overview of existing network-based data sets rather than contributing an additional data set.

Other recent papers also touch upon network-based data sets, yet with a different primary focus. Bhuyan et al. [31] present a comprehensive review of network anomaly detection. The authors describe nine existing data sets and analyze data sets which are used by existing anomaly detection methods. Similarly, Nisioti et al. [32] focus on unsupervised methods for intrusion detection and briefly refer to 12 existing network-based data sets. Yavanoglu and Aydos [33] analyze and compare the most commonly used data sets for intrusion detection. However, their review contains only seven data sets including other data sets like HTTP CSIC 2010 [34]. All in all, these works tend to have different research objectives and only touch upon network-based data sets marginally.

III. Data

Normally, network traffic is captured either in packet-based or flow-based format. Capturing network traffic on packet-level is usually done by mirroring ports on network devices. Packet-based data encompass complete payload information. Flow-based data are more aggregated and usually contain only metadata from network connections. Wheelus et al. highlight the distinction through an illustrative comparison: *"A good example of the difference between captured packet inspection and NetFlow would be viewing a forest by hiking through the forest as opposed to flying over the forest in a hot air balloon"* [35]. In this work, a third category (*other* data) is introduced. The category *other* has no standard format and varies for each data set.

A. Packet-based data

Packet-based data is commonly captured in pcap format and contains payload. Available metadata depends on the used network and transport protocols. There are many different protocols and the most important ones being TCP, UDP, ICMP and IP. Figure 1 illustrates the different headers. TCP is a reliable transport protocol and encompasses metadata like sequence number, acknowledgment number, TCP flags, or checksum values. UDP is a connection-less transport protocol and has a smaller header than TCP which contains only four fields, namely source port, destination port, length and checksum. In contrast to TCP and UDP, ICMP is a supporting protocol containing status messages and is thus even smaller. Normally, there is also an IP header available beside the header of the transport protocol. The IP header provides information such as source and destination IP addresses and is also shown in Figure 1.

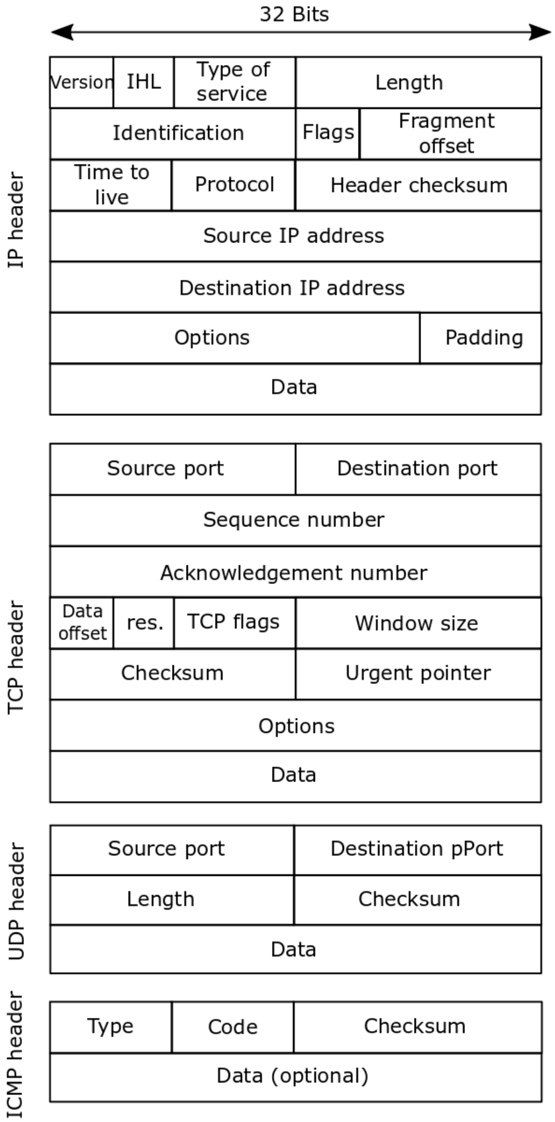


Fig. 1. IP, TCP, UDP and ICMP header after [36].

B. Flow-based data

Flow-based network data is a more condensed format which contains mainly meta information about network connections. Flow-based data aggregate all packets which share some properties within a time window into one flow and usually do not include any payload. The default five-tuple definition, i.e., source IP address, source port, destination IP address, destination port and transport protocol [37], is a widely used

TABLE I
ATTRIBUTES IN FLOW-BASED NETWORK TRAFFIC.

#	Attribute
1	Date first seen
2	Duration
3	Transport protocol
4	Source IP address
5	Source port
6	Destination IP address
7	Destination port
8	Number of transmitted bytes
9	Number of transmitted packets
10	TCP flags

standard for matching properties in flow-based data. Flows can appear in unidirectional or bidirectional format. The unidirectional format aggregates all packets from host *A* to host *B* which share the above mentioned properties into one flow. All packets from host *B* to host *A* are aggregated into another unidirectional flow. In contrast, a bidirectional flow summarizes all packets between hosts *A* and *B*, regardless of direction.

Typical flow-based formats are NetFlow [38], IPFIX [37], sFlow [39] and OpenFlow [40]. Table I gives an overview of typical attributes within flow-based network traffic. Depending on the specific flow format and flow exporter, additional attributes like bytes per second, bytes per packet, TCP flags of the first packet, or even the calculated entropy of the payload can be extracted.

Furthermore, it is possible to convert packet-based data to flow-based data (but not vice versa) with tools like nfdump[2] or YAF[3]. Readers interested in the differences between flow exporters may find additional details in [41], together with an analysis of how different flow exporters affect botnet classification.

C. Other data

This category includes all data sets that are neither purely packet-based nor flow-based. An example of this category might be flow-based data sets which have been enriched with additional information from packet-based data or host-based log files. The KDD CUP 1999 [42] data set is a well-known representative of this category. Each data point has network-based attributes like the number of transmitted source bytes or TCP flags, but has also host-based attributes like number of failed logins. As a consequence, each data set of this category has its own set of attributes. Since each data set must be analyzed individually, we do not make any general statements about available attributes.

IV. DATA SET PROPERTIES

To be able to compare different intrusion detection data sets side by side and to help researchers finding appropriate data sets for their specific evaluation scenario, it is necessary to define common properties as evaluation basis. Therefore, we explore typical data set properties that are used in the literature to assess intrusion detection data sets. The general concept FAIR [43] defines four principles that scholarly data should fulfill, namely *Findability*, *Accessibility*, *Interoperability* and *Reusability*. While concurring with this general concept, this work uses more detailed data set properties to provide a focused comparison of network-based intrusion detection data sets. Generally, different data sets emphasize different data set properties. For instance, the UGR'16 data set [29] emphasizes a long recording time to capture periodic effects while the ISCX data set [28] focuses on accurate labeling. Since we aim at investigating more general properties for network-based intrusion detection data sets, we try to unify and generalize properties used in literature rather than adopting all of them. For example, some approaches evaluate the presence of specific kind of attacks like DoS (Denial of Service) or Browser injections. The presence of certain attack types may be a relevant property for evaluating detection approaches for those specific attack types, but are meaningless for other approaches. Hence, we use the general property attacks which describes the presence of malicious network traffic (see Table III). Section V provides more details on the different attack types in the data sets together with a discussion of other particular properties.

We do not develop an evaluation score like Haider et al. [19] or Sharafaldin et al. [30] since we do not want to judge the importance of different data set properties. In our opinion, the importance of certain properties depends on the specific evaluation scenario and should not be generally judged in a survey. Rather, readers should be put in a position to find suitable data sets for their needs. Therefore, we group the data set properties discussed below in five categories to support systematic search. Figure 2 summarizes all data set properties and their value ranges.

A. General Information

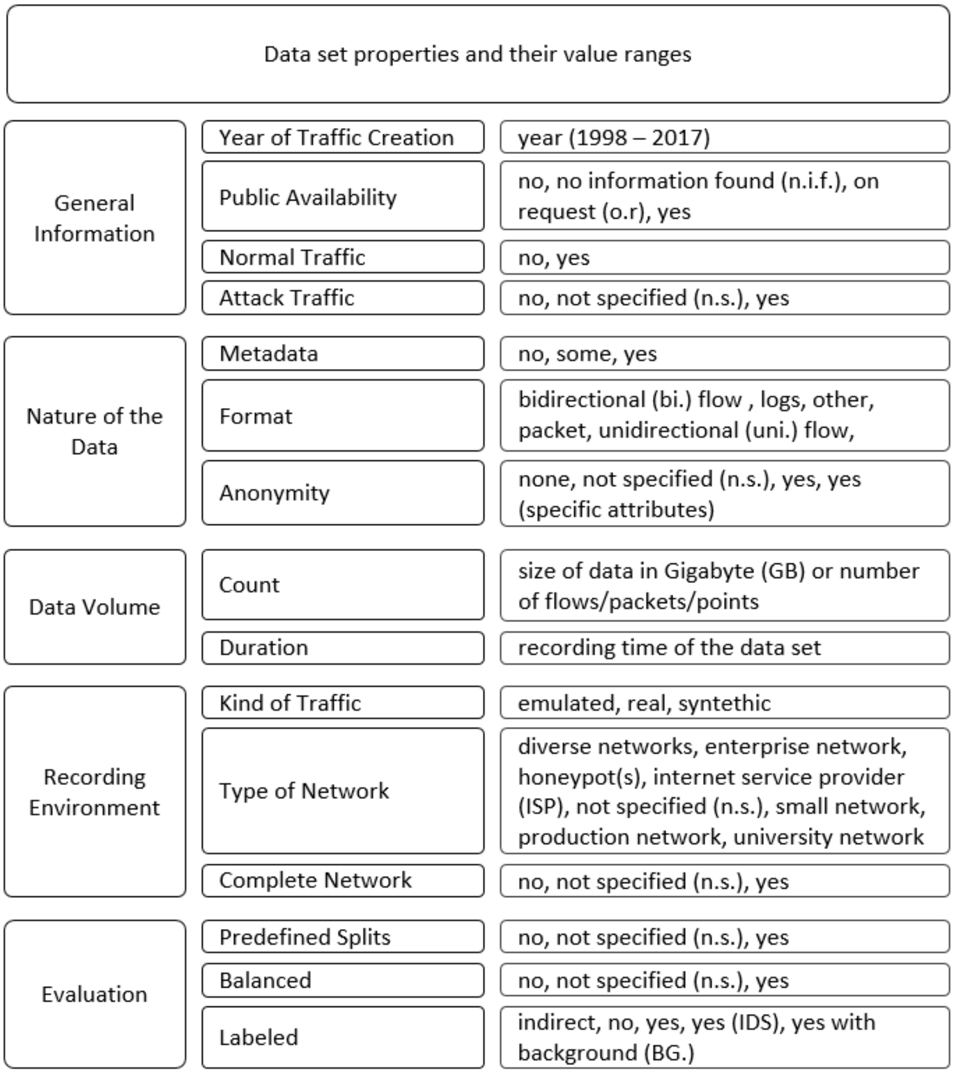
The following four properties reflect general information about the data set, namely the year of creation, availability, presence of normal and malicious network traffic.

1) Year of Creation: Since network traffic is subject to concept drift and new attack scenarios appear daily, the age of an intrusion detection data set plays an important role. This property describes the year of creation. The year in which the underlying network traffic of a data set was captured is more relevant for up-to-dateness than the year of its publication.

2) Public Availability: Intrusion detection data sets should be publicly available to serve as a basis for comparing different intrusion detection methods. Furthermore, the quality of data sets can only be checked by third parties if they are publicly available. Table III encompasses three different characteristics for this property: yes, o.r. (on request), and no. On request means that access will be granted after sending a message to the authors or the responsible person.

3) Normal User Behavior: This property indicates the availability of normal user behavior within a data set and takes the values yes or no. The value yes indicates that there is normal user behavior within the data set, but it does not make any statements about the presence of attacks. In general, the quality of an IDS is primarily determined by its attack

[2] https://github.com/phaag/nfdump
[3] https://tools.netsa.cert.org/yaf/



Data set properties and their value ranges

General Information	Year of Traffic Creation	year (1998 – 2017)
	Public Availability	no, no information found (n.i.f.), on request (o.r), yes
	Normal Traffic	no, yes
	Attack Traffic	no, not specified (n.s.), yes
Nature of the Data	Metadata	no, some, yes
	Format	bidirectional (bi.) flow , logs, other, packet, unidirectional (uni.) flow,
	Anonymity	none, not specified (n.s.), yes, yes (specific attributes)
Data Volume	Count	size of data in Gigabyte (GB) or number of flows/packets/points
	Duration	recording time of the data set
Recording Environment	Kind of Traffic	emulated, real, syntethic
	Type of Network	diverse networks, enterprise network, honeypot(s), internet service provider (ISP), not specified (n.s.), small network, production network, university network
	Complete Network	no, not specified (n.s.), yes
Evaluation	Predefined Splits	no, not specified (n.s.), yes
	Balanced	no, not specified (n.s.), yes
	Labeled	indirect, no, yes, yes (IDS), yes with background (BG.)

Fig. 2. Data set properties and their value ranges.

detection rate and false alarm rate. Therefore, the presence of normal user behavior is indispensable for evaluating an IDS. The absence of normal user behavior, however, does not make a data set unusable, but rather indicates that it has to be merged with other data sets or with real world network traffic. Such a merging step is often called overlaying or salting [44], [45].

4) Attack Traffic: IDS data sets should include various attack scenarios. This property indicates the presence of malicious network traffic within a data set and has the value yes if the data set contains at least one attack. Table IV provides additional information about the specific attack types.

B. Nature of Data

Properties of this category describe the format of the data sets and the presence of meta information.

1) Metadata: Content-related interpretation of packet-based and flow-based network traffic is difficult for third parties. Therefore, data sets should come along with metadata to provide additional information about the network structure, IP addresses, attack scenarios and so on. This property indicates the presence of additional metadata.

2) Format: Network intrusion detection data sets appear in different formats. We roughly divide them into three formats (see Section III). (1) Packet-based network traffic (e.g. pcap) contains network traffic with payload. (2) Flow-based network traffic (e.g. NetFlow) contains only meta information about network connections. (3) Other types of data sets may contain, e.g., flow-based traces with additional attributes from packet-based data or even from host-based log files.

3) Anonymity: Frequently, intrusion detection data sets may not be published due to privacy reasons or are only available in anonymized form. This property indicates if data is anonymized and which attributes are affected. The value none in Table III indicates that no anonymization has been performed. The value yes (IPs) means that IP addresses are either anonymized or removed from the data set. Similarly, yes (payload) indicates that payload information is anonymized or removed from packet-based network traffic.

C. Data Volume

Properties in this category characterize data sets in terms of volume and duration.

1) Count: The property count describes a data set's size as either the number of contained packets/flows/points or the physical size in Gigabyte (GB).

2) Duration: Data sets should cover network traffic over a long time for capturing periodical effects (e.g., daytime vs. night or weekday vs. weekend) [29]. The property duration provides the recording time of each data set.

D. Recording Environment

Properties in this category delineate the network environment and conditions in which the data sets are captured.

1) Kind of Traffic: The property Kind of Traffic describes three possible origins of network traffic: real, emulated, or synthetic. Real means that real network traffic was captured within a productive network environment. Emulated means that real network traffic was captured within a test bed or emulated network environment. Synthetic means that the network traffic was created synthetically (e.g., through a traffic generator) and not captured by a real (or virtual) network device.

2) Type of Network: Network environments in small and medium-sized companies are fundamentally different from internet service providers (ISP). As a consequence, different environments require different security systems and evaluation data sets should be adapted to the specific environment. This property describes the underlying network environment in which the respective data set was created.

3) Complete Network: This property is adopted from Sharafaldin et al. [30] and indicates if the data set contains the complete network traffic from a network environment with several hosts, router and so on. If the data set contains only network traffic from a single host (e.g., honeypot) or only some protocols from the network traffic (e.g., exclusively SSH traffic), the value is set to no.

E. Evaluation

The following properties are related to the evaluation of intrusion detection methods using network-based data sets. More precisely, the properties denote the availability of predefined subsets, the data set's balance, and the presence of labels.

1) Predefined Splits: Sometimes it is difficult to compare the quality of different IDS, even if they are evaluated on the same data set. In that case, it must be clarified whether the same subsets are used for training and evaluation. This property provides the information if a data set comes along with predefined subsets for training and evaluation.

2) Balanced: Often, machine learning and data mining methods are used for anomaly-based intrusion detection. In the training phase of such methods (e.g., decision tree classifiers), data sets should be balanced with respect to their class labels. Consequently, data sets should contain the same number of data points from each class (normal and attack). Real-world network traffic, however, is not balanced and contains more normal user behavior than attack traffic. This property indicates if data sets are balanced with respect to their class labels. Imbalanced data sets should be balanced by appropriate preprocessing before data mining algorithms are used. He and Garcia [46] provide a good overview of learning from imbalanced data.

3) Labeled: Labeled data sets are necessary for training supervised methods and for evaluating supervised as well as unsupervised intrusion detection methods. This property denotes if data sets are labeled or not. If there are at least the two classes normal and attack, this property is set to yes. Possible values in this property are: yes, yes with BG. (yes with background), yes (IDS), indirect, and no. Yes with background means that there is a third class background. Packets, flows, or data points which belong to the class background could be normal or attack. Yes (IDS) means that some kind of intrusion detection system was used to create the data set's labels. Some labels of the data set might be wrong since an IDS might be imperfect. Indirect means that the data set has no explicit labels, but labels can be created on one's own from additional log files.

V. Data Sets

In our opinion, the data set properties *Labeled* and *Format* are the most decisive properties when searching for adequate network-based data sets. The intrusion detection method (supervised or unsupervised) determines if labels are necessary or not and which kind of data is required (packet, flow or *other*). Therefore, Table II provides a classification of all investigated network-based data sets with respect to these two properties. A more detailed overview of network-based intrusion detection data sets with respect to the data set properties of Section IV is given in Table III. The presence of specific attack scenarios is an important aspect when searching for a network-based data set. Therefore, Table III indicates the presence of attack traffic while Table IV provides details on specific attacks within a data set. Papers on data sets describe attacks on different abstraction levels. Vasudevan et al. [47], for instance, characterized attack traffic within their data set (SSENET-2011) as follows: *"Nmap, Nessus, Angry IP scanner, Port Scanner, Metaploit, Backtrack OS, LOIC, etc., were some of the attack tools used by the participants to launch the attacks."*. In contrast, Ring et al. specify the number and different types of executed port scans in their CIDDS-002 data set [27]. Consequently, the abstraction level of attack descriptions may vary in Table IV. A detailed description of all attack types is beyond the scope of this work. Rather, we refer interested readers to the open access paper *"From Intrusion Detection to an Intrusion Response System: Fundamentals, Requirements, and Future Directions"* by Anwar et al. [48].

Further, some data sets are modifications or combinations of others. Figure 3 shows the interrelationships among several well-known data sets.

Network-based data sets in alphabetical order

AWID [49]. AWID is a publicly available data set[4] which is focused on 802.11 networks. Its creators used a small network

[4] http://icsdweb.aegean.gr/awid/index.html

TABLE II
DECISION SUPPORT TABLE FOR FINDING APPROPRIATE NETWORK-BASED DATA SETS. SOME DATA SETS LIKE CTU-13 PROVIDE SEVERAL DATA FORMATS AND APPEAR IN SEVERAL COLUMNS. (+) INDICATES THAT THE DATA SET IS PUBLICLY AVAILABLE. (?) INDICATES THAT WE WERE NOT ABLE TO FIND THE DATA SET. (-) INDICATES THAT THE DATA SET IS NOT PUBLICLY AVAILABLE.

	Format		
Labeled	packet	flow	other
yes	(+) Botnet (+) CIC DoS (+) CICIDS 2017 (+) DARPA (+) DDoS 2016 (+) ISCX 2012 (+) ISOT (+) NDSec-1 (+) NGIDS-DS (+) TRAbID (+) TUIDS (+) UNSW-NB15	(+) CICIDS 2017 (+) CIDDS-001 (+) CIDDS-002 (+) ISCX 2012 (+) TUIDS (+) Twente	(+) AWID (+) KDD CUP 99 (+) Kyoto 2006+ (+) NSL-KDD (+) UNSW-NB 15
			(?) PU-IDS (?) SSENET-2011 (?) SSENET-2014
	(-) IRSC	(-) IRSC	(-) SANTA
yes with BG	(+) CTU-13	(+) CTU-13 (+) UGR'16	
yes (IDS)		(?) PUF	
indirect		(+) SSHCure	
no	(+) Booters (+) CDX (+) LBNL	(+) Kent 2016 (+) UNIBS (+) Unified Host and Network	

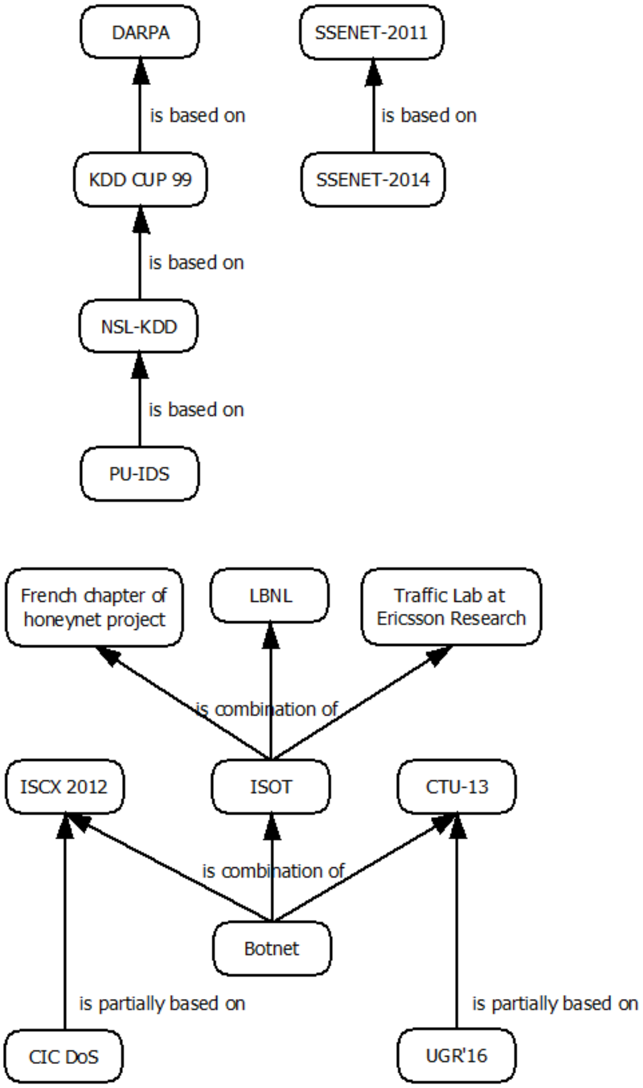


Fig. 3. Relationships between the data sets in Table III.

environment (11 clients) and captured WLAN traffic in packet-based format. In one hour, 37 million packets were captured. 156 attributes are extracted from each packet. Malicious network traffic was generated by executing 16 specific attacks against the 802.11 network. AWID is labeled and split into a training and a test subset.

Booters [50]. Booters are Distributed Denial of Service (DDoS) attacks offered as a service by criminals. Santanna et. al [50] published a data set which includes traces of nine different booter attacks which were executed against a null-routed IP address within their network environment. The resulting data set is recorded in packet-based format and consists of more than 250GB of network traffic. Individual packets are not labeled, but the different booter attacks are split into different files. The data set is publicly available[5], but names of booters are anonymized for privacy reasons.

Botnet [5]. The Botnet data set is a combination of existing data sets and is publicly available[6]. The creators of Botnet used the overlay methodology of [44] to combine (parts of) the ISOT [57], ISCX 2012 [28] and CTU-13 [3] data sets. The resulting data set contains various botnets and normal user behavior. The Botnet data set is divided into a 5.3 GB training subset and a 8.5 GB test subset, both in packet-based format.

CIC DoS [51]. CIC DoS is a data set from the Canadian Institute for Cybersecurity and is publicly available[7]. The authors' intention was to create an intrusion detection data set with application layer DoS attacks. Therefore, the authors executed eight different DoS attacks on the application layer. Normal user behavior was generated by combining the resulting traces with attack-free traffic from the ISCX 2012 [28] data set. The resulting data set is available in packet-based format and contains 24 hours of network traffic.

CICIDS 2017 [22]. CICIDS 2017 was created within an emulated environment over a period of 5 days and contains network traffic in packet-based and bidirectional flow-based format. For each flow, the authors extracted more than 80 attributes and provide additional metadata about IP addresses and attacks. Normal user behavior is executed through scripts. The data set contains a wide range of attack types like SSH brute force, heartbleed, botnet, DoS, DDoS, web and infiltration attacks. CICIDS 2017 is publicly available[8].

CIDDS-001 [21]. The CIDDS-001 data set was captured within an emulated small business environment in 2017, contains four weeks of unidirectional flow-based network

[5] https://www.simpleweb.org/wiki/index.php

[6] http://www.unb.ca/cic/datasets/botnet.html

[7] http://www.unb.ca/cic/datasets/dos-dataset.html

[8] http://www.unb.ca/cic/datasets/ids-2017.html

TABLE III
OVERVIEW OF NETWORK-BASED DATA SETS.

	General Information				**Nature of the Data**			**Data Volume**		**Recording Environment**			**Evaluation**		
Data Set	Year of Traffic Creation	Public Avail.	Normal Traffic	Attack Traffic	Meta-data	Format	Anonymity	Count	Duration	Kind of Traffic	Type of Network	Compl. Network	Predef. Splits	Balanced	Labeled
AWID [49]	2015	o.r.	yes	yes	yes	other	none	37M packets	1 hour	emulated	small network	yes	yes	no	yes
Booters [50]	2013	yes	no	yes	no	packet	yes	250GB packets	2 days	real	small network	no	no	no	no
Botnet [5]	2010/2014	yes	yes	yes	yes	packet	none	14GB packets	n.s.	emulated	diverse networks	yes	yes	no	yes
CIC DoS [51]	2012/2017	yes	yes	yes	no	packet	none	4.6GB packets	24 hours	emulated	small network	yes	no	no	yes
CICIDS 2017 [22]	2017	yes	yes	yes	yes	packet, bi. flow	none	3.1M flows	5 days	emulated	small network	yes	no	no	yes
CIDDS-001 [21]	2017	yes	yes	yes	yes	uni. flow	yes (IPs)	32M flows	28 days	emulated and real	small network	yes	no	no	yes
CIDDS-002 [27]	2017	yes	yes	yes	yes	uni. flow	yes (IPs)	15M flows	14 days	emulated	small network	yes	no	no	yes
CDX [52]	2009	yes	yes	yes	yes	packet	none	14GB packets	4 days	real	small network	yes	no	no	no
CTU-13 [3]	2013	yes	yes	yes	yes	uni. and bi. flow, paket	yes (payload)	81M flows	125 hours	real	university network	yes	no	no	yes with BG.
DARPA [53], [54]	1998/99	yes	yes	yes	yes	packet, logs	none	n.s.	7/5 weeks	emulated	small network	yes	yes	no	yes
DDoS 2016 [55]	2016	yes	yes	yes	no	packet	yes (IPs)	2.1M packets	n.s.	synthetic	n.s.	n.s.	no	no	yes
IRSC [56]	2015	no	yes	yes	no	packet, flow	n.s.	n.s.	n.s.	real	production network	yes	n.s.	n.s.	yes
ISCX 2012 [28]	2012	yes	yes	yes	yes	packet, bi. flow	none	2M flows	7 days	emulated	small network	yes	no	no	yes
ISOT [57]	2010	yes	yes	yes	yes	packet	none	11GB packets	n.s.	emulated	small network	yes	no	no	yes
KDD CUP 99 [42]	1998	yes	yes	yes	no	other	none	5M points	n.s.	emulated	small network	yes	yes	no	yes
Kent 2016 [58], [59]	2016	yes	yes	n.s.	no	uni. flow, logs	yes (IPs, Ports, date)	130M flows	58 days	real	enterprise network	yes	no	no	no
Kyoto 2006+ [60]	2006 to 2009	yes	yes	yes	no	other	yes (IPs)	93M points	3 years	real	honeypots	no	no	no	yes
LBNL [61]	2004 / 2005	yes	yes	yes	no	packet	yes	160M packets	5 hours	real	enterprise network	yes	no	no	no
NDSec-1 [62]	2016	o.r.	no	yes	no	packet, logs	none	3.5M packets	n.s.	emulated	small network	yes	no	no	yes
NGIDS-DS [19]	2016	yes	yes	yes	no	packet, logs	none	1M packets	5 days	emulated	small network	yes	no	no	yes
NSL-KDD [63]	1998	yes	yes	yes	no	other	none	150k points	n.s.	emulated	small network	yes	yes	no	yes
PU-IDS [64]	1998	n.i.f.	yes	yes	no	other	none	200k points	n.s.	synthetic	small network	yes	no	no	yes
PUF [65]	2018	n.i.f.	yes	yes	no	uni. flow	yes (IPs)	300k flows	3 days	real	university network	no	no	no	yes (IDS)
SANTA [35]	2014	no	yes	yes	no	other	yes (payload)	n.s.	n.s.	real	ISP	yes	n.s.	no	yes
SSENET-2011 [47]	2011	n.i.f.	yes	yes	no	other	none	n.s.	4 hours	emulated	small network	yes	no	no	yes
SSENET-2014 [66]	2011	n.i.f.	yes	yes	no	other	none	200k points	4 hours	emulated	small network	yes	yes	yes	yes
SSHCure [67]	2013 / 2014	yes	yes	yes	no	uni. and bi. flow, logs	yes (IPs)	2.4GB flows (compressed)	2 months	real	university network	yes	no	no	indirect
TRAbID [68]	2017	yes	yes	yes	no	packet	yes (IPs)	460M packets	8 hours	emulated	small network	yes	yes	no	yes
TUIDS [69], [70]	2011 / 2012	o.r.	yes	yes	no	packet, bi. flow	none	250k flows	21 days	emulated	medium network	yes	yes	no	yes
Twente [71]	2008	yes	no	yes	yes	uni. flow	yes (IPs)	14M flows	6 days	real	honeypot	no	no	no	yes
UGR'16 [29]	2016	yes	yes	yes	some	uni. flows	yes (IPs)	16900M flows	4 months	real	ISP	yes	yes	no	yes with BG.
UNIBS [72]	2009	o.r.	yes	no	no	flow	yes (IPs)	79k flows	3 days	real	university network	yes	no	no	no
Unified Host and Network [73]	2017	yes	yes	n.s.	no	bi. flows, logs	yes (IPs and date)	150GB flows (compressed)	90 days	real	enterprise network	yes	no	no	no
UNSW-NB15 [20]	2015	yes	yes	yes	yes	packet, other	none	2M points	31 hours	emulated	small network	yes	yes	no	yes

n.s. = not specified, n.i.f. = no information found, uni. flow = unidirectional flow, bi. flow = bidirectional flow, yes with BG. = yes with background labels

TABLE IV
ATTACKS WITHIN THE NETWORK-BASED DATA SETS OF TABLE III. SPECIFIC ATTACK INFORMATION (E.G. NAME OF THE EXECUTED BOTNET) AND USED TOOLS ARE PROVIDED IN ROUND BRACKETS IF AVAILABLE.

Data Set	Attacks
AWID [49]	Popular attacks on 802.11 (e.g. authentication request, ARP flooding, injection, probe request)
Booters [50]	9 different DDoS attacks
Botnet [5]	botnets (Menti, Murlo, Neris, NSIS, Rbot, Sogou, Strom, Virut, Zeus)
CIC DoS [51]	Application layer DoS attacks (executed through ddossim, Goldeneye, hulk, RUDY, Slowhttptest, Slowloris)
CICIDS 2017 [22]	botnet (Ares), cross-site-scripting, DoS (executed through Hulk, GoldenEye, Slowloris, and Slowhttptest), DDoS (executed through LOIC), heartbleed, infiltration, SSH brute force, SQL injection
CIDDS-001 [21]	DoS, port scans (ping-scan, SYN-Scan), SSH brute force
CIDDS-002 [27]	port scans (ACK-Scan, FIN-Scan, ping-Scan, UDP-Scan, SYN-Scan)
CDX [52]	not specified
CTU-13 [3]	botnets (Menti, Murlo, Neris, NSIS, Rbot, Sogou, Virut)
DARPA [53], [54]	DoS, privilege escalation (remote-to-local and user-to-root), probing
DDoS 2016 [55]	DDoS (HTTP flood, SIDDOS, smurf ICMP flood, UDP flood)
IRSC [56]	n.s.
ISCX 2012 [28]	Four attack scenarios (1: Infiltrating the network from the inside; 2: HTTP DoS; 3: DDoS using an IRC botnet; 4: SSH brute force)
ISOT [57]	botnet (Storm, Waledac)
KDD CUP 99 [42]	DoS, privilege escalation (remote-to-local and user-to-root), probing
Kent 2016 [58], [59]	not specified
Kyoto 2006+ [60]	Various attacks against honeypots (e.g. backscatter, DoS, exploits, malware, port scans, shellcode)
LBNL [61]	port scans
NDSec-1 [62]	botnet (Citadel), brute force (against FTP, HTTP and SSH), DDoS (HTTP floods, SYN flooding and UDP floods), exploits, probe, spoofing, SSL proxy, XSS/SQL injection
NGIDS-DS [19]	backdoors, DoS, exploits, generic, reconnaissance, shellcode, worms
NSL-KDD [63]	DoS, privilege escalation (remote-to-local and user-to-root), probing
PU-IDS [64]	DoS, privilege escalation (remote-to-local and user-to-root), probing
PUF [65]	DNS attacks
SANTA [35]	(D)DoS (ICMP flood, RUDY, SYN flood), DNS amplification, heartbleed, port scans
SSENET-2011 [47]	DoS (executed through LOIC), port scans (executed through Angry IP Scanner, Nessus, Nmap), various attack tools (e.g. metasploit)
SSENET-2014 [66]	botnet, flooding, privilege escalation, port scans
SSHCure [67]	SSH attacks
TRAbID [68]	DoS (HTTP flood, ICMP flood, SMTP flood, SYN flood, TCP keepalive), port scans (ACK-Scan, FIN-Scan, NULL-Scan, OS Fingerprinting, Service Fingerprinting, UDP-Scan, XMAS-Scan)
TUIDS [69], [70]	botnet (IRC), DDoS (Fraggle flood, Ping flood, RST flood, smurf ICMP flood, SYN flood, UDP flood), port scans (e.g. FIN-Scan, NULL-Scan, UDP-Scan, XMAS-Scan), coordinated port scan, SSH brute force
Twente [71]	Attacks against a honeypot with three open services (FTP, HTTP, SSH)
UGR'16 [29]	botnet (Neris), DoS, port scans, SSH brute force, spam
UNIBS [72]	none
Unified Host and Network [73]	not specified
UNSW-NB15 [20]	backdoors, DoS, exploits, fuzzers, generic, port scans, reconnaissance, shellcode, spam, worms

traffic, and comes along with a detailed technical report with additional information. As special feature, the data set encompasses an external server which was attacked in the internet. In contrast to honeypots, this server was also regularly used by the clients from the emulated environment. Normal and malicious user behavior was executed through python scripts which are publicly available on GitHub[9]. These scripts allow an ongoing generation of new data sets and can be used by other researches. The CIDDS-001 data set is publicly available[10] and contains SSH brute force, DoS and port scan attacks as well as several attacks captured from the wild.

CIDDS-002 [27]. CIDDS-002 is a port scan data set which is created based on the scripts of CIDDS-001. The data set contains two weeks of unidirectional flow-based network traffic within an emulated small business environment. CIDDS-002 contains normal user behavior as well as a wide range of different port scan attacks. A technical report provides additional meta information about the data set where external IP addresses are anonymized. The data set is publicly available[11].

CDX [52]. Sangster el al. [52] propose a concept to create network-based data sets from network warfare competitions and discuss the advantages and disadvantages of such an approach comprehensively. The CDX data set contains network traffic of a four day network warfare competition in 2009. The traffic is recorded in packet-based format and is publicly available[12]. CDX contains normal user behaviour as well as several types of attacks. An additional plan describes metadata about the network structure and IP addresses, but the individual packets are not labeled. Further, host-based log files and warnings from an IDS are available.

CTU-13 [3]. The CTU-13 data set was captured in the year 2013 and is available in three formats: packet, unidirectional flow, and bidirectional flow[13]. It was captured in a university network and distinguishes 13 scenarios containing different botnet attacks. Additional information about infected hosts is provided at the website. Traffic was labeled using a three stage approach. In the first stage, all traffic to and from infected hosts is labeled as botnet. In the second stage, traffic which matches specific filters is labeled as normal. Remaining traffic is labeled as background. Consequently, background traffic could be normal or malicious. The authors recommend a split of their data set into training and test subsets [3].

DARPA [53], [54], [74]. The DARPA 1998/99 data sets are the most popular data sets for intrusion detection and were created at the MIT Lincoln Lab within an emulated network environment. The DARPA 1998 and DARPA 1999 data sets contain seven and, respectively, five weeks of network traffic in packet-based format, including various kinds of attacks like DoS, buffer overflow, port scans, or rootkits. Additional information as well as download links can be found at the website[14]. In spite (or because) of their wide distribution, the

[9] https://github.com/markusring/CIDDS

[10] http://www.hs-coburg.de/cidds

[11] http://www.hs-coburg.de/cidds

[12] https://www.usma.edu/crc/sitepages/datasets.aspx

[13] https://mcfp.weebly.com/the-ctu-13-dataset-a-labeled-dataset-with-botnet-normal-and-b html

[14] https://www.ll.mit.edu/ideval/docs/index.html

data sets are often criticized for artificial attack injections or the large amount of redundancy [63], [75].

DDoS 2016 [55]. Alkasassbeh et al. [55] published a packet-based data set which was created using the network simulator NS2 in 2016. Detailed information about the simulated network environment is not available. The DDoS 2016 data set focuses on different types of DDoS attacks. In addition to normal network traffic, the data set contains four different types of DDoS attacks: UDP flood, smurf, HTTP flood, and SIDDOS. The data set contains 2.1 million packets and can be downloaded at researchgate[15].

IRSC [56]. The IRSC data set was recorded in 2015, using an inventive approach. Real network traffic with normal user behavior and attacks from the internet were captured. In addition to that, additional attacks were run manually. The IDS SNORT[16] and manual inspection were used for labeling. Since the data set is not publicly available due to privacy concerns, we are not able to fill all properties in Table III.

ISCX 2012 [28]. The ISCX data set was created in 2012 by capturing traffic in an emulated network environment over one week. The authors used a dynamic approach to generate an intrusion detection data set with normal as well as malicious network behavior. So-called α profiles define attack scenarios while β profiles characterize normal user behavior like writing e-mails or browsing the web. These profiles are used to create a new data set in packet-based and bidirectional flow-based format. The dynamic approach allows an ongoing generation of new data sets. ISCX can be downloaded at the website[17] and contains various types of attacks like SSH brute force, DoS or DDoS.

ISOT [57]. The ISOT data set was created in 2010 by combining normal network traffic from Traffic Lab at Ericsson Research in Hungary [76] and the Lawrence Berkeley National Lab (LBNL) [61] with malicious network traffic from the French chapter of the honeynet project[18]. ISOT was used for detecting P2P botnets [57]. The resulting data set is publicly available[19] and contains 11 GB of packet-based data in pcap format.

KDD CUP 99 [42]. KDD CUP 99 is based on the DARPA data set and among the most widespread data sets for intrusion detection. Since it is neither in standard packet-, nor in flow-based format, it belongs to category *other*. The data set contains basic attributes about TCP connections and high-level attributes like number of failed logins, but no IP addresses. KDD CUP 99 encompasses more than 20 different types of attacks (e.g. DoS or buffer overflow) and comes along with an explicit test subset. The data set includes 5 million data points and can be downloaded freely[20].

Kent 2016 [58], [59]. This data set was captured over 58 days at the Los Alamos National Laboratory network. It contains around 130 million flows of unidirectional flow-based network traffic as well as several host-based log files. Network traffic is heavily anonymized for privacy reasons. The data set is not labeled and can be downloaded at the website[21].

Kyoto 2006+ [60]. Kyoto 2006+ is a publicly available honeypot data set[22] which contains real network traffic, but includes only a small amount and a small range of realistic normal user behavior. Kyoto 2006+ is categorized as *other* since the IDS Bro[23] was used to convert packet-based traffic into a new format called sessions. Each session comprises 24 attributes, 14 out of which characterize statistical information inspired by the KDD CUP 99 data set. The remaining 10 attributes are typical flow-based attributes like IP addresses (in anonymized form), ports, or duration. A label attribute indicates the presence of attacks. Data were captured over three years. As a consequence of that unusually long recording period, the data set contains about 93 million sessions.

LBNL [61]. Research on intrusion detection data sets often refers to the LBNL data set. Thus, for the sake of completeness, this data set is also added to the list. The creation of the LNBL data set was mainly motivated by analyzing characteristics of network traffic within enterprise networks, rather than publishing intrusion detection data. According to its creators, the data set might still be used as background traffic for security researchers as it contains almost exclusively normal user behavior. The data set is not labeled, but anonymized for privacy reasons, and contains more than 100 hours of network traffic in packet-based format. The data set can be downloaded at the website[24].

NDSec-1 [62]. The NDSec-1 data set is remarkable since it is designed as an attack composition for network security. According to the authors, this data set can be reused to salt existing network traffic with attacks using overlay methodologies like [44]. NDSec-1 is publicly available on request[25] and was captured in packet-based format in 2016. It contains additional syslog and windows event log information. The attack composition of NDSec-1 encompasses botnet, brute force attacks (against FTP, HTTP and SSH), DoS (HTTP flooding, SYN flooding, UDP flooding), exploits, port scans, spoofing, and XSS/SQL injection.

NGIDS-DS [19]. The NGIDS-DS data set contains network traffic in packet-based format as well as host-based log files. It was generated in an emulated environment, using the IXIA Perfect Storm tool to generate normal user behavior as well as attacks from seven different attack families (e.g. DoS or worms). Consequently, the quality of the generated data depends primarily on the IXIA Perfect Storm hardware[26]. The labeled data set contains approximately 1 million packets and is publicly available[27].

NSL-KDD [63]. NSL-KDD enhances the KDD CUP 99. A major criticism against the KDD CUP 99 data set is the

[15] https://www.researchgate.net/publication/292967044_Dataset-_Detecting_Distributed_Denial_of_Service_Attacks_Using_Data_Mining_Techniques

[16] https://www.snort.org/

[17] http://www.unb.ca/cic/datasets/ids.html

[18] http://honeynet.org/chapters/france

[19] https://www.uvic.ca/engineering/ece/isot/datasets/

[20] http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html

[21] https://csr.lanl.gov/data/cyber1/

[22] http://www.takakura.com/Kyoto_data/

[23] https://www.bro.org/

[24] http://icir.org/enterprise-tracing/

[25] http://www2.hs-fulda.de/NDSec/NDSec-1/

[26] https://www.ixiacom.com/products/perfectstorm

[27] https://research.unsw.edu.au/people/professor-jiankun-hu

large amount of redundancy [63]. Therefore, the authors of NSL-KDD removed duplicates from the KDD CUP 99 data set and created more sophisticated subsets. The resulting data set contains about 150,000 data points and is divided into predefined training and test subsets for intrusion detection methods. NSL-KDD uses the same attributes as KDD CUP 99 and belongs to the category *other*. Yet, it should be noted that the underlying network traffic of NSL-KDD dates back to the year 1998. The data set is publicly available[28].

PU-IDS [64]. The PU-IDS data set is a derivative of the NSL-KDD data set. The authors developed a generator which extracts statistics of an input data set and uses these statistics to generate new synthetic instances. As a consequence, the work of Singh et al. [64] could be seen as a traffic generator to create PU-IDS which contains about 200,000 data points and has the same attributes and format as the NSL-KDD data set. As NSL-KDD is based on KDD CUP 1999 which in turn is extracted from DARPA 1998, the year of creation is set to 1998 since the input for the traffic generator was captured back then.

PUF [65]. Recently, Sharma et al. [65] published the flow-based PUF data set which was captured over three days within a campus network and contains exclusively DNS connections. 38,120 out of a total of 298,463 unidirectional flows are malicious while the remaining ones reflect normal user activity. All flows are labeled using logs of an intrusion prevention system. For privacy reasons, IP addresses are removed from the data set. The authors intend to make PUF publicly available.

SANTA [35]. The SANTA data set was captured within an ISP environment and contains real network traffic. The network traffic is labeled through an exhaustive manual procedure and stored in a so-called session-based format. This data format is similar to NetFlow but enriched with additional attributes which are calculated by using information from packet-based data. The authors spent much effort on the generation of additional attributes which should enhance intrusion detection methods. SANTA is not publicly available.

SSENet-2011 [47]. SSENet-2011 was captured within an emulated environment over four hours. It contains several attacks like DoS or port scans. Browsing activities of participants generated normal user behavior. Each data point is characterized by 24 attributes. The data set belongs to the category *other* since the tool Tstat was used to extract adjusted data points from packet-based traffic. We found no information about public availability.

SSENet-2014 [66]. SSENet-2014 is created by extracting attributes from the packet-based files of SSENet-2011 [47]. Thus, like SSENet-2011, the data set is categorized as *other*. The authors extracted 28 attributes for each data point which describe host-based and network-based attributes. The created attributes are in line with KDD CUP 1999. SSENet-2014 contains 200,000 labeled data points and is divided into a training and test subnet. SSENet-2014 is the only known data set with a balanced training subset. Again, no information on public availability could be found.

SSHCure [67]. Hofstede et al. [67] propose SSHCure, a tool for SSH attack detection. To evaluate their work, the authors captured two data sets (each with a period of one month) within a university network. The resulting data sets are publicly available[29] and contain exclusively SSH network traffic. The flow-based network traffic is not directly labeled. Instead, the authors provide additional host-based logs files which may be used to check if SSH login attempts were successful or not.

TRAbID [68]. Viegas et al. proposed the TRAbID database [68] in 2017. This database contains 16 different scenarios for evaluating IDS. Each scenario was captured within an emulated environment (1 honeypot server and 100 clients). In each scenario, the traffic was captured for a period of 30 minutes and some attacks were executed. To label the network traffic, the authors used the IP addresses of the clients. All clients were Linux machines. Some clients exclusively performed attacks while most of the clients exclusively handled normal user requests to the honeypot server. Normal user behavior includes HTTP, SMTP, SSH and SNMP traffic while malicious network traffic encompasses port scans and DoS attacks. TRAbID is publicly available[30].

TUIDS [69], [70]. The labeled TUIDS data set can be divided into three parts: TUIDS Intrusion data set, TUIDS coordinated scan data set and TUIDS DDoS data set. As the names already indicate, the data sets contain normal user behavior and primarily attacks like port scans or DDoS. Data were generated within an emulated environment which contains around 250 clients. Traffic was captured in packet-based and bidirectional flow-based format. Each subset spans a period of seven days and all three subsets contain around 250,000 flows. Unfortunately, the link[31] to the data set in the original publication seems to be outdated. However, the authors respond to e-mail requests.

Twente [71]. Sperotto et al. [71] published one of the first flow-based intrusion detection data sets in 2008. This data set spans six days of traffic involving a honeypot server which offers web, FTP, and SSH services. Due to this approach, the data set contains only network traffic from the honeypot and nearly all flows are malicious without normal user behavior. The authors analyzed log files and traffic in packet-based format for labeling the flows of this data set. The data set is publicly available[32] and IP addresses were removed due to privacy concerns.

UGR'16 [29]. UGR'16 is a unidirectional flow-based data set. Its focus lies on capturing periodic effects in an ISP environment. Thus, it spans a period of four months and contains 16,900 million unidirectional flows. IP addresses are anonymized and the flows are labeled as normal, background, or attack. The authors explicitly executed several attacks (botnet, DoS, and port scans) within that data set. The corresponding flows are labeled as attacks and some other attacks were identified and manually labeled as attack. Injected normal user behavior and traffic which matches certain patterns are

[28] http://www.unb.ca/cic/datasets/nsl.html

[29] https://www.simpleweb.org/wiki/index.php

[30] https://secplab.ppgia.pucpr.br/trabid

[31] http://agnigarh.tezu.ernet.in/~dkb/resources.html

[32] https://www.simpleweb.org/wiki/index.php

labeled as normal. However, most of the traffic is labeled as background which could be normal or an attack. The data set is publicly available[33].

UNIBS 2009 [72]. Like LBNL [61], the UNIBS 2009 data set was not created for intrusion detection. Since UNIBS 2009 is referenced in other work, it is still added to the list. Gringoli et al. [72] used the data set to identify applications (e.g. web browsers, Skype or mail clients) based on their flow-based network traffic. UNIBS 2009 contains around 79,000 flows without malicious behavior. Since the labels just describe the application protocols of the flows, network traffic is not categorized as normal or attack. Consequently, the property label in the categorization scheme is set to *no*. The data set is publicly available[34].

Unified Host and Network Data Set [73]. This data set contains host and network-based data which were captured within a real environment, the LANL (Los Alamos National Laboratory) enterprise network. For privacy reasons, attributes like IP addresses and timestamps were anoynmized in bidirectional flow-based network traffic files. The network traffic was collected for a period of 90 days and has no labels. The data set is publicly available[35].

UNSW-NB15 [20]. The UNSW-NB15 data set encompasses normal and malicious network traffic in packet-based format which was created using the IXIA Perfect Storm tool in a small emulated environment over 31 hours. It contains nine different families of attacks like backdoors, DoS, exploits, fuzzers, or worms. The data set is also available in flow-based format with additional attributes. UNSW-NB15 comes along with predefined splits for training and test. The data set includes 45 distinct IP addresses and is publicly available[36].

VI. Other Data Sources

Besides network-based data sets, there are some other data sources for packet-based and flow-based network traffic. In the following, we shortly discuss data repositories and traffic generators.

A. Data Repositories

Besides traditional data sets, several data repositories can be found on the internet. Since type and structure of those repositories differ greatly, we abstain from a tabular comparison. Instead, we give a brief textual overview in alphabetical order. Repositories have been checked on 26 February 2019 with respect to actuality.

AZSecure[37]. AZSecure is a repository of network data at the University of Arizona for use by the research community. It includes various data sets in pcap, arff and other formats some of which are labeled, while other are not. AZSecure encompasses, among others, the CTU-13 data set [3] or the Unified Host and Network Data Set [73]. The repository is managed and contains some recent data sets.

CAIDA[38]. CAIDA collects different types of data sets, with varying degree of availability (public access or on request), and provides a search engine. Generally, a form needs to be filled out to gain access to some of the public data sets. Additionally, most network-based data sets can exclusively be requested through an IMPACT (see below) login since CAIDA supports IMPACT as Data Provider. The repository is managed and updated with new data.

Contagiodump[39]. Contagiodump is a blog about malware dumps. There are several posts each year and the last post was on 20th March 2018. The website contains, among other things, a collection of pcap files from malware analysis.

covert.io[40]. Covert.io is a blog about security and machine learning by Jason Trost. The blog maintains different lists of tutorials, GitHub repositories, research papers and other blogs concerning security, big data, and machine learning, but also a collection of various security-based data resources[41]. The latest entry was posted on August 14, 2017 by Jason Trost.

DEF CON CTF Archive[42]. DEF CON is a popular annual hacker convention. The event includes a capture the flag (CTF) competition where every team has to defend their own network against the other teams whilst simultaneously hacking the opponents' networks. The competition is typically recorded and available in packet-based format on the website. Given the nature of the competition, the recorded data almost exclusively contain attack traffic and little normal user behavior. The website is current and updated annually with new data from the CTF competitions.

IMPACT[43]. IMPACT Cyber Trust, formerly known as PREDICT, is a community of data providers, cyber security researchers as well as coordinators. IMPACT is administrated and up-to-date. A data catalog is given on the site to browse the data sets provided by the community. The data providers are (among others) DARPA, the MIT Lincoln Laboratory, or the UCSD - Center for Applied Internet Data Analysis (CAIDA). However, the data sets can only be downloaded with an account that may be requested exclusively by researchers from eight selected countries approved by the US Department of Homeland Security. As Germany is not among the approved locations, no further statements about the data sets can be made.

Internet Traffic Archive[44]. The Internet Traffic Archive is a repository of internet traffic traces sponsored by ACM SIGCOMM. The list includes four extensively anonymized packet-based traces. In particular, the payload has been removed, all timestamps are relative to the first packet, and IP addresses have been changed to numerical representations. The packet-based data sets were captured more than 20 years ago and can be downloaded without restriction.

Kaggle[45]. Kaggle is an online platform for sharing and

[33] https://nesg.ugr.es/nesg-ugr16/index.php

[34] http://netweb.ing.unibs.it/~ntw/tools/traces/)

[35] https://csr.lanl.gov/data/2017.html

[36] https://www.unsw.adfa.edu.au/unsw-canberra-cyber/cybersecurity/ADFA-NB15-Datasets/

[37] https://www.azsecure-data.org/other-data.html

[38] http://www.caida.org/data/overview/

[39] http://contagiodump.blogspot.com/

[40] http://www.covert.io

[41] http://www.covert.io/data-links/

[42] https://www.defcon.org/html/links/dc-ctf.html

[43] https://www.impactcybertrust.org/

[44] http://ita.ee.lbl.gov/html/traces.html

[45] https://www.kaggle.com/

publishing data sets. The platform contains security-based data sets like KDD CUP 99 and has a search function. It allows registered users also to upload and explore data analysis models.

Malware Traffic Analysis[46]. Malware Traffic Analysis is a repository which contains blog posts and exercises related to network traffic analysis, e.g. identifying malicious activities. Exercises come along with packet-based network traffic which is indirectly labeled through the provided answers to the exercises. Downloadable files are secured with a password which can be obtained from the website. The repository is recent and new blog posts are issued almost daily.

Mid-Atlantic CCDC[47]. Similar to DEFCON CTF, MACCDC is an annual competition hosted by the US National CyberWatch Center where the captured packet-based traffic of the competitions is made available. Teams have to assure that services provided by their network are not interrupted in any way. Similar to the DEFCON CTF archives, MACCDC data contain almost exclusively attack traffic and little normal user behavior. The latest competition took place in 2018.

MAWILab[48]. The MAWILab repository contains a huge amount of network traffic over a long time which is captured at a link between USA and Japan. For each day since 2007, the repository contains a 15 minute trace in packet-based format. For privacy reasons, IP addresses are anonymized and packet payloads are omitted. The captured network traffic is labeled using different anomaly detection methods [77].

MWS[49]**.** The anti malware engineering workshop (MWS) is an annual workshop about malware in Japan. The workshop comes along with several MWS data sets which contain packet-based network data as well as host-based log files. However, the data sets are only shared within the MWS community which consists of researches in industry and academia in Japan [78]. The latest workshop took place in 2018.

NETRECSEC[50]. NETRECSEC maintains a comprehensive list of publicly available pcap files on the internet. Similar to SecRepo, NETRECSEC refers to many repositories mentioned in this work, but also incorporates additional sources like honeypot dumps or CTF events. Its up-to-dateness can only be judged indirectly as NETRECSEC also refers to data traces from the year 2018.

OpenML[51]. OpenML is an update-to-date platform for sharing machine learning data sets. It contains also security-based data sets like KDD CUP 99. The platform has a search function and comes along with other possibilities like creating scientific tasks.

RIPE Data Repository[52]. The RIPE data repository hosts a number of data sets. Yet, no new data sets have been included for several years. To obtain access, users need to create an account and accept the terms and conditions of the data sets. The repository also mirrors some data available from the Waikato Internet Traffic Storage (see below).

SecRepo[53]. SecRepo lists different samples of security related data and is maintained by Mike Sconzo. The list is divided in the following categories: Network, Malware, System, File, Password, Threat Feeds and Other. The very detailed list contains references to typical data sets like DARPA, but also to many repositories (e.g. NETRECSEC). The website was last updated on November 20, 2018.

Simple Web[54]. Simple Web provides a database collection and information on network management tutorials and software. The repository includes traces in different formats like packet or flow-based network traffic. It is hosted by the University of Twente, maintained by members of the DACS (Design and Analysis of Communication Systems) group, and updated with new results from this group.

UMassTraceRepository[55]. UMassTraceRepository provides the research community with several traces of network traffic. Some of these traces have been collected by the suppliers of the archive themselves while others have been donated. The archive includes 19 packet-based data sets from different sources. The most recent data sets were captured in 2018.

VAST Challenge[56]. The IEEE Visual Analytics Science and Technology (VAST) challenge is an annual contest with the goal of advancing the field of visual analytics through competition. In some challenges, network traffic data were provided for contest tasks. For instance, the second mini challenge of the VAST 2011 competition involved an IDS log consisting of packet-based network traffic in pcap format. A similar setup was used in a follow-up VAST challenge in 2012. Furthermore, a VAST challenge in 2013 deals with flow-based network traffic.

WITS: Waikato Internet Traffic Storage[57]. This website aims to list all internet traces possessed by the WAND research group. The data sets are typically available in packet-based format and free to download from the Waikato servers. However, the repository has not been updated for a long time.

B. Traffic Generators

Another source of network traffic for intrusion detection research are traffic generators. Traffic generators are models which create synthetic network traffic. In most cases, traffic generators use user-defined parameters or extract basic properties of real network traffic to create new synthetic network traffic. While data sets and data repositories provide fixed data, traffic generators allow the generation of network traffic which can be adapted to certain network structures.

For instance, the traffic generators FLAME [79] and ID2T [80] use real network traffic as input. This input traffic should serve as a baseline for normal user behavior. Then, FLAME and ID2T add malicious network traffic by editing

[46] http://malware-traffic-analysis.net/
[47] http://maccdc.org/
[48] http://www.fukuda-lab.org/mawilab/
[49] https://www.iwsec.org/mws/2018/en.html
[50] http://www.netresec.com/?page=PcapFiles
[51] https://www.openml.org/home
[52] https://labs.ripe.net/datarepository
[53] http://www.secrepo.com/
[54] https://www.simpleweb.org/wiki/index.php/
[55] http://traces.cs.umass.edu/
[56] http://vacommunity.org/tiki-index.php
[57] https://wand.net.nz/wits/catalogue.php

values of input traffic or by injecting synthetic flows under consideration of typical attack patterns. Siska et al. [81] present a graph-based flow generator which extracts traffic templates from real network traffic. Then, their generator uses these traffic templates in order to create new synthetic flow-based network traffic. Ring et al. [82] adapted GANs for generating synthetic network traffic. The authors use Improved Wasserstein Generative Adversarial Networks (WGAN-GP) to create flow-based network traffic. The WGAN-GP is trained with real network traffic and learns traffic characteristics. After training, the WGAN-GP is able to create new synthetic flow-based network traffic with similar characteristics. Erlacher and Dressler's traffic generator GENESIDS [83] generates HTTP attack traffic based on user defined attack descriptions. There are many additional traffic generators which are not discussed here for the sake of brevity. Besides those traffic generators, there are many other traffic generators which are not discussed here. Instead, we refer to Molnár et al. [84] for an overview of traffic generators.

Brogi et al. [85] come up with another idea that in some sense resembles traffic generators. Starting out from the problem of sharing data sets due to privacy concerns, they present Moirai, a framework which allows users to share complete scenarios instead of data sets. The idea behind Moirai is to replay attack scenarios in virtual machines such that users can generate data on the fly.

A third approach - which is also categorized into the larger context of traffic generators - are frameworks which support users to label real network traffic. Rajasinghe et al. present such a framework called INSecS-DCS [86] which captures network traffic at network devices or uses already captured network traffic in pcap files as input. Then, INSecS-DCS divides the data stream into time windows, extracts data points with appropriate attributes, and labels the network traffic based on a user-defined attacker IP address list. Consequently, the focus of INSecS-DCS is on labeling network traffic and on extracting meaningful attributes. Aparicio-Navarro et al. [87] present an automatic data set labeling approach using an unsupervised anomaly-based IDS. Since no IDS is able to classify each data point to the correct class, the authors take some middle ground to reduce the number of false positives and true negatives. The IDS assigns belief values to each data point for the classes normal and attack. If the difference between the belief values for these two classes is smaller than a predefined threshold, the data point is removed from the data set. This approach increases the quality of the labels, but may discard the most interesting data points of the data set.

VII. Observations and Recommendations

Labeled data sets are inevitable for training supervised data mining methods like classification algorithms and helpful for the evaluation of supervised as well as unsupervised data mining methods. Consequently, labeled network-based data sets can be used to compare the quality of different NIDS with each other. In any case, however, the data sets must be representative to be suitable for those tasks. The community is aware of the importance of realistic network-based data, and this survey shows that there are many sources for such data (data sets, data repositories, and traffic generators). Furthermore, this work establishes a collection of data set properties as a basis for comparing available data sets and for identifying suitable data sets, given specific evaluation scenarios.

In the following, we discuss some aspects concerning the use of available data sets and the creation of new data sets.

Perfect data set: The ever-increasing number of attack scenarios, accompanied by new and more complex software and network structures, leads to the requirement that data sets should contain up-to-date and real network traffic. Since there is no perfect IDS, labeling of data points should be checked manually rather than being done exclusively by an IDS. Consequently, the perfect network-based data set is up-to-date, correctly labeled, publicly available, contains real network traffic with all kinds of attacks and normal user behavior as well as payload, and spans a long time. Such a data set, however, does not exist and will (probably) never be created. If privacy concerns could be satisfied and real-world network traffic (in packet-based format) with all kind of attacks could be recorded over a sufficiently long time, accurate labeling of such traffic would be very time-consuming. As a consequence, the labeling process would take so much time that the data set is slightly outdated since new attack scenarios appear continuously. However, several available data sets satisfy some properties of a perfect data set. Besides, most applications do not require a perfect data set - a data set which satisfies certain properties is often sufficient. For instance, there is no need that a data set contains all types of attacks when evaluating a new port scan detection algorithm, or there is no need for complete network configuration when evaluating the security of a specific server. Therefore, we hope that this work supports researchers to find the appropriate data set for their specific evaluation scenario.

Use of several data sets: As mentioned above, no perfect network-based data set exists. However, this survey shows that there are several data sets (and other data sources) available for packet- and flow-based network traffic. Therefore, we recommend users to evaluate their intrusion detection methods with more than one data set in order to avoid over-fitting to a certain data set, reduce the influence of artificial artifacts of a certain data set, and evaluate their methods in a more general context. In addition to that, Hofstede et al. [88] show that flow-based network traffic differs between lab environments and production networks. Therefore, another approach could be to use both, emulated respectively synthetic data sets and real world network traffic to emphasize these points.

In order to ensure reproducibility for third parties, we recommend evaluating intrusion detection methods with at least one publicly available data set.

Further, we would like to give a general recommendation for the use of the CICIDS 2017, CIDDS-001, UGR'16 and UNSW-NB15 data sets. These data sets may be suitable for general evaluation settings. CICIDS 2017 and UNSW-NB15 contain a wide range of attack scenarios. CIDDS-001 contains detailed metadata for deeper investigations. UGR'16 stands out by the huge amount of flows. However, it should be considered that this recommendation reflects our personal views. The recommendation does **not** imply that other data sets are inappropriate. For instance, we only refrain to include the more widespread CTU-13 and ISCX 2012 data sets in our recommendation due to their increasing age. Further, other data sets like AWID or Botnet are better suited for certain evaluation scenarios.

Predefined Subsets: Furthermore, we want to make a note on the evaluation of anomaly-based NIDS. Machine learning and data mining methods often use so-called 10-fold cross-validation [89]. This method divides the data set into ten equal-sized subsets. One subset is used for testing and the other nine subsets are used for training. This procedure is repeated ten times, such that every subset has been used once for testing. However, this straight-forward splitting of data sets makes only limited sense for intrusion detection. For instance, the port scan data set CIDDS-002 [27] contains two weeks of network traffic in flow-based format. Each port scan within this data set may cause thousands of flows. Using 10-fold cross-validation would lead to the situation that probably some flows of each attack appear in the training data set. Thus, attack detection in test data is facilitated and generalization is not properly evaluated.

In that scenario, it would be better to train on week1 and test on week2 (and vice versa) for the CIDDS-002 data set. Defining subsets on that approach may also consider the impact of concept drift in network traffic over time. Another approach for creating suitable subsets might be to split the whole data set based on traffic characteristics like source IP addresses. However, such subsets must be well designed to preserve the basic network structures of the data set. For instance, a training data set with exclusively source IP addresses which represent clients and no severs would be inappropriate.

Based on these observations, we recommend creating meaningful training and test splits with respect to the application domain IT security. Therefore, benchmark data sets should be published with predefined splits for training and test to facilitate comparisons of different approaches evaluated on the same data.

Closer collaboration: This study shows (see Section V) that many data sets have been published in the last few years and the community works on creating new intrusion detection data sets continuously. Further, the community could benefit from closer collaboration and a single generally accepted platform for sharing intrusion detection data sets without any access restrictions. For instance, Cermak et al. [90] work on establishing such a platform for sharing intrusion detection data sets. Likewise, Ring et al. [21] published their scripts for emulating normal user behavior and attacks such that they can be used and improved by third parties. A short summary of all mentioned data sets and data repositories can be found at our website [58] and we intend to update this website with upcoming network-based data sets.

Standard formats: Most network-based intrusion detection approaches require standard input data formats and cannot handle preprocessed data. Further, it is questionable if data sets from category *other* (Section III-C) can be calculated in real time which may affect their usefulness in NIDS. Therefore, we suggest providing network-based data sets in standard packet-based or flow-based formats as they are captured in real network environments. Simultaneously, many anomaly-based approaches (e.g., [91] or [92]) achieve high detection rates in data sets from the category *other* which is an indicator that the calculated attributes are promising for intrusion detection. Therefore, we recommend publishing both, the network-based data sets in a standard format and the scripts for transforming the data sets to other formats. Such an approach would have two advantages. First, users may decide if they want to transfer data sets to other formats and a larger number of researchers could use the corresponding data sets. Second, the scripts could also be applied to future data sets.

Anonymization: Anonymization is another important issue since this may complicate the analysis of network-based data sets. Therefore, it should be carefully evaluated which attributes have to be discarded and which attributes may be published in anonymous form. Various authors demonstrate the effectiveness of using only small parts of payload. For example, Mahoney [93] proposes an intrusion detection method which uses the first 48 bytes of each packet starting with the IP header. The flow exporter YAF [94] allows the creation of such attributes by extracting the first *n* bytes of payload or by calculating the entropy of the payload. Generally, there are several methods for anonymization. For example, Xu et al. [95] propose a prefix-preserving IP address anonymization technique. Tcpmkpub [96] is an anonymization tool for packet-based network traffic which allows the anonymization of some attributes like IP addresses and also computes new values for header checksums. We refer to Kelly et al. [97] for a more comprehensive review of anonymization techniques for network-based data.

Publication: We recommend the publication of network-based data sets. Only publicly available data sets can be used by third parties and thus serve as a basis for evaluating NIDS. Likewise, the quality of data sets can only be checked by third parties if they are publicly available. Last but not least, we recommend the publication of additional metadata such that third parties are able to analyze the data and their results in more detail.

VIII. Summary

Labeled network-based data sets are necessary for training and evaluating NIDS. This paper provides a literature survey of available network-based intrusion detection data sets. To this end, standard network-based data formats are analyzed in more detail. Further, 15 properties are identified that may be used to assess the suitability of data sets. These properties are grouped into five categories: General Information, Nature of the Data, Data Volume, Recording Environment and Evaluation.

The paper's main contribution is a comprehensive overview of 34 data sets which points out the peculiarities of each data set. Thereby a particular focus was placed on attack scenarios within the data sets and their interrelationships. In addition, each data set assessed with respect to the properties of the categorization scheme developed in the first step. This

[58] http://www.dmir.uni-wuerzburg.de/datasets/nids-ds

detailed investigation aims to support readers to identify data sets for their purposes. The review of data sets shows that the research community has noticed a lack of publicly available network-based data sets and tries to overcome this shortage by publishing a considerable number of data sets over the last few years. Since several research groups are active in this area, additional intrusion detection data sets and improvements can be expected soon.

As further sources for network traffic, traffic generators and data repositories are discussed in Section VI. Traffic generators create synthetic network traffic and can be used to create adapted network traffic for specific scenarios. Data repositories are collections of different network traces on the internet. Compared to the data sets in Section V, data repositories often provide limited documentation, non-labeled data sets or network traffic of specific scenarios (e.g., exclusively FTP connections). However, these data sources should be taken into account when searching for suitable data, especially for specialized scenarios. Finally, we discussed some observations and recommendations for the use and generation of network-based intrusion detection data sets. We encourage users to evaluate their methods on several data sets to avoid over-fitting to a certain data set and to reduce the influence of artificial artifacts of a certain data set. Further, we advocate data sets in standard formats including predefined training and test subsets. Overall, there probably won't be a perfect data set, but there are many very good data sets available and the community could benefit from closer collaboration.

Acknowledgment

M.R. and S.W. were supported by the BayWISS Consortium Digitisation. S.W. is additionally funded by the Bavarian State Ministry of Science and Arts in the framework of the Centre Digitisation.Bavaria (ZD.B).

References

[1] V. Chandola, E. Eilertson, L. Ertoz, G. Simon, V. Kumar, Data Mining for Cyber Security, in: A. Singhal (Ed.), Data Warehousing and Data Mining Techniques for Computer Security, 1st Edition, Springer, 2006, pp. 83–107.

[2] M. Rehák, M. Pechoucek, K. Bartos, M. Grill, P. Celeda, V. Krmicek, CAMNEP: An intrusion detection system for high-speed networks, Progress in Informatics 5 (5) (2008) 65–74. doi:10.2201/NiiPi.2008.5.7.

[3] S. Garcia, M. Grill, J. Stiborek, A. Zunino, An empirical comparison of botnet detection methods, Computers & Security 45 (2014) 100–123. doi:10.1016/j.cose.2014.05.011.

[4] M. Ring, S. Wunderlich, D. Grüdl, D. Landes, A. Hotho, A Toolset for Intrusion and Insider Threat Detection, in: I. Palomares, H. Kalutarage, Y. Huang (Eds.), Data Analytics and Decision Support for Cybersecurity: Trends, Methodologies and Applications, Springer, 2017, pp. 3–31. doi:10.1007/978-3-319-59439-2_1.

[5] E. B. Beigi, H. H. Jazi, N. Stakhanova, A. A. Ghorbani, Towards Effective Feature Selection in Machine Learning-Based Botnet Detection Approaches, in: IEEE Conference on Communications and Network Security, IEEE, 2014, pp. 247–255. doi:10.1109/CNS.2014.6997492.

[6] M. Stevanovic, J. M. Pedersen, An analysis of network traffic classification for botnet detection, in: IEEE International Conference on Cyber Situational Awareness, Data Analytics and Assessment (CyberSA), IEEE, 2015, pp. 1–8. doi:10.1109/CyberSA.2015.7361120.

[7] J. Wang, I. C. Paschalidis, Botnet Detection Based on Anomaly and Community Detection, IEEE Transactions on Control of Network Systems 4 (2) (2017) 392–404. doi:10.1109/TCNS.2016.2532804.

[8] C. Yin, Y. Zhu, S. Liu, J. Fei, H. Zhang, An Enhancing Framework for Botnet Detection Using Generative Adversarial Networks, in: International Conference on Artificial Intelligence and Big Data (ICAIBD), 2018, pp. 228–234. doi:10.1109/ICAIBD.2018.8396200.

[9] S. Staniford, J. A. Hoagland, J. M. McAlerney, Practical automated detection of stealthy portscans, Journal of Computer Security 10 (1-2) (2002) 105–136.

[10] J. Jung, V. Paxson, A. W. Berger, H. Balakrishnan, Fast Portscan Detection Using Sequential Hypothesis Testing, in: IEEE Symposium on Security & Privacy, IEEE, 2004, pp. 211–225. doi:10.1109/SECPRI.2004.1301325.

[11] A. Sridharan, T. Ye, S. Bhattacharyya, Connectionless Port Scan Detection on the Backbone, in: IEEE International Performance Computing and Communications Conference, IEEE, 2006, pp. 10–19. doi:10.1109/.2006.1629454.

[12] M. Ring, D. Landes, A. Hotho, Detection of slow port scans in flow-based network traffic, PLOS ONE 13 (9) (2018) 1–18. doi:10.1371/journal.pone.0204507.

[13] A. Sperotto, R. Sadre, P.-T. de Boer, A. Pras, Hidden Markov Model modeling of SSH brute-force attacks, in: International Workshop on Distributed Systems: Operations and Management, Springer, 2009, pp. 164–176. doi:10.1007/978-3-642-04989-7_13.

[14] L. Hellemons, L. Hendriks, R. Hofstede, A. Sperotto, R. Sadre, A. Pras, SSHCure: A Flow-Based SSH Intrusion Detection System, in: International Conference on Autonomous Infrastructure, Management and Security (IFIP), Springer, 2012, pp. 86–97. doi:10.1007/978-3-642-30633-4_11.

[15] M. Javed, V. Paxson, Detecting Stealthy, Distributed SSH Brute-Forcing, in: ACM SIGSAC Conference on Computer & Communications Security, ACM, 2013, pp. 85–96. doi:10.1145/2508859.2516719.

[16] M. M. Najafabadi, T. M. Khoshgoftaar, C. Kemp, N. Seliya, R. Zuech, Machine Learning for Detecting Brute Force Attacks at the Network Level, in: International Conference on Bioinformatics and Bioengineering (BIBE), IEEE, 2014, pp. 379–385. doi:10.1109/BIBE.2014.73.

[17] R. Sommer, V. Paxson, Outside the Closed World: On Using Machine Learning For Network Intrusion Detection, in: IEEE Symposium on Security and Privacy, IEEE, 2010, pp. 305–316. doi:10.1109/SP.2010.25.

[18] M. Małowidzki, P. Berezinski, M. Mazur, Network Intrusion Detection: Half a Kingdom for a Good Dataset, in: NATO STO SAS-139 Workshop, Portugal, 2015.

[19] W. Haider, J. Hu, J. Slay, B. Turnbull, Y. Xie, Generating realistic intrusion detection system dataset based on fuzzy qualitative modeling, Journal of Network and Computer Applications 87 (2017) 185–192. doi:10.1016/j.jnca.2017.03.018.

[20] N. Moustafa, J. Slay, UNSW-NB15: A Comprehensive Data Set for Network Intrusion Detection Systems, in: Military Communications and Information Systems Conference (MilCIS), IEEE, 2015, pp. 1–6. doi:10.1109/MilCIS.2015.7348942.

[21] M. Ring, S. Wunderlich, D. Grüdl, D. Landes, A. Hotho, Flow-based benchmark data sets for intrusion detection, in: European Conference on Cyber Warfare and Security (ECCWS), ACPI, 2017, pp. 361–369.

[22] I. Sharafaldin, A. H. Lashkari, A. A. Ghorbani, Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization, in: International Conference on Information Systems Security and Privacy (ICISSP), 2018, pp. 108–116. doi:10.5220/0006639801080116.

[23] G. Creech, J. Hu, Generation of a New IDS Test Dataset: Time to Retire the KDD Collection, in: IEEE Wireless Communications and Networking Conference (WCNC), IEEE, 2013, pp. 4487–4492. doi:10.1109/WCNC.2013.6555301.

[24] T. R. Glass-Vanderlan, M. D. Iannacone, M. S. Vincent, R. A. Bridges, et al., A Survey of Intrusion Detection Systems Leveraging Host Data, arXiv preprint arXiv:1805.06070.

[25] R. Koch, M. Golling, G. D. Rodosek, Towards Comparability of Intrusion Detection Systems: New Data Sets, in: TERENA Networking Conference, Vol. 7, 2014.

[26] J. O. Nehinbe, A critical evaluation of datasets for investigating IDSs and IPSs Researches, in: IEEE International Conference on Cybernetic Intelligent Systems (CIS), IEEE, 2011, pp. 92–97. doi:10.1109/CIS.2011.6169141.

[27] M. Ring, S. Wunderlich, D. Grüdl, D. Landes, A. Hotho, Creation of Flow-Based Data Sets for Intrusion Detection, Journal of Information Warfare 16 (2017) 40–53.

[28] A. Shiravi, H. Shiravi, M. Tavallaee, A. A. Ghorbani, Toward developing a systematic approach to generate benchmark datasets for

intrusion detection, Computers & Security 31 (3) (2012) 357–374. doi:10.1016/j.cose.2011.12.012.

[29] G. Maciá-Fernández, J. Camacho, R. Magán-Carrión, P. García-Teodoro, R. Therón, UGR'16: A New Dataset for the Evaluation of Cyclostationarity-Based Network IDSs, Computers & Security 73 (2018) 411–424. doi:10.1016/j.cose.2017.11.004.

[30] I. Sharafaldin, A. Gharib, A. H. Lashkari, A. A. Ghorbani, Towards a Reliable Intrusion Detection Benchmark Dataset, Software Networking 2018 (1) (2018) 177–200. doi:10.13052/jsn2445-9739.2017.009.

[31] M. H. Bhuyan, D. K. Bhattacharyya, J. K. Kalita, Network Anomaly Detection: Methods, Systems and Tools, IEEE Communications Surveys & Tutorials 16 (1) (2014) 303–336. doi:10.1109/SURV.2013.052213.00046.

[32] A. Nisioti, A. Mylonas, P. D. Yoo, V. Katos, From Intrusion Detection to Attacker Attribution: A Comprehensive Survey of Unsupervised Methods, IEEE Communications Surveys Tutorials 20 (4) (2018) 3369–3388. doi:10.1109/COMST.2018.2854724.

[33] O. Yavanoglu, M. Aydos, A Review on Cyber Security Datasets for Machine Learning Algorithms, in: IEEE International Conference on Big Data, IEEE, 2017, pp. 2186–2193. doi:10.1109/BigData.2017.8258167.

[34] C. T. Giménez, A. P. Villegas, G. Á. Marañón, HTTP data set CSIC 2010, (Date last accessed 22-June-2018) (2010).

[35] C. Wheelus, T. M. Khoshgoftaar, R. Zuech, M. M. Najafabadi, A Session Based Approach for Aggregating Network Traffic Data - The SANTA Dataset, in: IEEE International Conference on Bioinformatics and Bioengineering (BIBE), IEEE, 2014, pp. 369–378. doi:10.1109/BIBE.2014.72.

[36] A. S. Tanenbaum, D. Wetherall, Computer Networks, 5th Edition, Pearson, 2011.

[37] B. Claise, Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of IP Traffic Flow Information, RFC 5101 (2008).

[38] B. Claise, Cisco Systems NetFlow Services Export Version 9, RFC 3954 (2004).

[39] P. Phaal, sFlow Specification Version 5 (2004).
URL https://sflow.org/sflow_version_5.txt

[40] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, J. Turner, OpenFlow: Enabling Innovation in Campus Networks, ACM SIGCOMM Computer Communication Review 38 (2) (2008) 69–74. doi:10.1145/1355734.1355746.

[41] F. Haddadi, A. N. Zincir-Heywood, Benchmarking the Effect of Flow Exporters and Protocol Filters on Botnet Traffic Classification, IEEE Systems Journal 10 (4) (2016) 1390–1401. doi:10.1109/JSYST.2014.2364743.

[42] S. Stolfo, (Date last accessed 22-June-2018). [link].
URL http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html

[43] M. D. Wilkinson, M. Dumontier, I. J. Aalbersberg, G. Appleton, M. Axton, A. Baak, N. Blomberg, J.-W. Boiten, L. B. da Silva Santos, P. E. Bourne, et al., The FAIR Guiding Principles for scientific data management and stewardship, Scientific Data 3. doi:10.1038/sdata.2016.18.

[44] A. J. Aviv, A. Haeberlen, Challenges in Experimenting with Botnet Detection Systems, in: Conference on Cyber Security Experimentation and Test (CEST), USENIX Association, Berkeley, CA, USA, 2011.

[45] Z. B. Celik, J. Raghuram, G. Kesidis, D. J. Miller, Salting Public Traces with Attack Traffic to Test Flow Classifiers, in: Workshop on Cyber Security Experimentation and Test (CSET), 2011.

[46] H. He, E. A. Garcia, Learning from Imbalanced Data, IEEE Transactions on Knowledge and Data Engineering 21 (9) (2009) 1263–1284. doi:10.1109/TKDE.2008.239.

[47] A. R. Vasudevan, E. Harshini, S. Selvakumar, SSENet-2011: A Network Intrusion Detection System dataset and its comparison with KDD CUP 99 dataset, in: Second Asian Himalayas International Conference on Internet (AH-ICI), 2011, pp. 1–5. doi:10.1109/AHICI.2011.6113948.

[48] S. Anwar, J. Mohamad Zain, M. F. Zolkipli, Z. Inayat, S. Khan, B. Anthony, V. Chang, From Intrusion Detection to an Intrusion Response System: Fundamentals, Requirements, and Future Directions, Algorithms 10 (2) (2017) 39.

[49] C. Kolias, G. Kambourakis, A. Stavrou, S. Gritzalis, Intrusion Detection in 802.11 Networks: Empirical Evaluation of Threats and a Public Dataset, IEEE Communications Surveys Tutorials 18 (1) (2016) 184–208. doi:10.1109/COMST.2015.2402161.

[50] J. J. Santanna, R. van Rijswijk-Deij, R. Hofstede, A. Sperotto, M. Wierbosch, L. Z. Granville, A. Pras, Booters - An analysis of DDoS-as-a-service attacks, in: IFIP/IEEE International Symposium on Integrated Network Management (IM), 2015, pp. 243–251. doi:10.1109/INM.2015.7140298.

[51] H. H. Jazi, H. Gonzalez, N. Stakhanova, A. A. Ghorbani, Detecting HTTP-based application layer DoS attacks on web servers in the presence of sampling, Computer Networks 121 (2017) 25–36. doi:10.1016/j.comnet.2017.03.018.

[52] B. Sangster, T. O'Connor, T. Cook, R. Fanelli, E. Dean, C. Morrell, G. J. Conti, Toward Instrumenting Network Warfare Competitions to Generate Labeled Datasets, in: Workshop on Cyber Security Experimentation and Test (CSET), 2009.

[53] R. P. Lippmann, D. J. Fried, I. Graf, J. W. Haines, K. R. Kendall, D. McClung, D. Weber, S. E. Webster, D. Wyschogrod, R. K. Cunningham, et al., Evaluating Intrusion Detection Systems : The 1998 DARPA Off-line Intrusion Detection Evaluation, in: DARPA Information Survivability Conference and Exposition (DISCEX), Vol. 2, IEEE, 2000, pp. 12–26. doi:10.1109/DISCEX.2000.821506.

[54] R. Lippmann, J. W. Haines, D. J. Fried, J. Korba, K. Das, The 1999 DARPA Off-Line Intrusion Detection Evaluation, Computer Networks 34 (4) (2000) 579–595. doi:10.1016/S1389-1286(00)00139-0.

[55] M. Alkasassbeh, G. Al-Naymat, A. Hassanat, M. Almseidin, Detecting Distributed Denial of Service Attacks Using Data Mining Techniques, International Journal of Advanced Computer Science and Applications (IJACSA) 7 (1) (2016) 436–445.

[56] R. Zuech, T. M. Khoshgoftaar, N. Seliya, M. M. Najafabadi, C. Kemp, A New Intrusion Detection Benchmarking System, in: International Florida Artificial Intelligence Research Society Conference (FLAIRS), AAAI Press, 2015, pp. 252–256.

[57] S. Saad, I. Traore, A. Ghorbani, B. Sayed, D. Zhao, W. Lu, J. Felix, P. Hakimian, Detecting P2P Botnets through Network Behavior Analysis and Machine Learning, in: International Conference on Privacy, Security and Trust (PST), IEEE, 2011, pp. 174–180. doi:10.1109/PST.2011.5971980.

[58] A. D. Kent, Comprehensive, Multi-Source Cyber-Security Events, Los Alamos National Laboratory (2015). doi:10.17021/1179829.

[59] A. D. Kent, Cybersecurity Data Sources for Dynamic Network Research, in: Dynamic Networks in Cybersecurity, Imperial College Press, 2015, pp. 37–65. doi:10.1142/9781786340757_0002.

[60] J. Song, H. Takakura, Y. Okabe, M. Eto, D. Inoue, K. Nakao, Statistical Analysis of Honeypot Data and Building of Kyoto 2006+ Dataset for NIDS Evaluation, in: Workshop on Building Analysis Datasets and Gathering Experience Returns for Security, ACM, 2011, pp. 29–36. doi:10.1145/1978672.1978676.

[61] R. Pang, M. Allman, M. Bennett, J. Lee, V. Paxson, B. Tierney, A First Look at Modern Enterprise Traffic, in: ACM SIGCOMM Conference on Internet Measurement (IMC), USENIX Association, Berkeley, CA, USA, 2005, pp. 15–28.

[62] F. Beer, T. Hofer, D. Karimi, U. Bühler, A new Attack Composition for Network Security, in: 10. DFN-Forum Kommunikationstechnologien, Gesellschaft für Informatik eV, 2017, pp. 11–20.

[63] M. Tavallaee, E. Bagheri, W. Lu, A. A. Ghorbani, A detailed analysis of the KDD CUP 99 data set, in: IEEE Symposium on Computational Intelligence for Security and Defense Applications, 2009, pp. 1–6. doi:10.1109/CISDA.2009.5356528.

[64] R. Singh, H. Kumar, R. Singla, A Reference Dataset for Network Traffic Activity Based Intrusion Detection System, International Journal of Computers Communications & Control 10 (3) (2015) 390–402. doi:10.15837/ijccc.2015.3.1924.

[65] R. Sharma, R. Singla, A. Guleria, A New Labeled Flow-based DNS Dataset for Anomaly Detection: PUF Dataset, Procedia Computer Science 132 (2018) 1458–1466, international Conference on Computational Intelligence and Data Science. doi:10.1016/j.procs.2018.05.079.

[66] S. Bhattacharya, S. Selvakumar, SSENet-2014 Dataset: A Dataset for Detection of Multiconnection Attacks, in: International Conference on Eco-friendly Computing and Communication Systems (ICECCS), IEEE, 2014, pp. 121–126. doi:10.1109/Eco-friendly.2014.100.

[67] R. Hofstede, L. Hendriks, A. Sperotto, A. Pras, SSH compromise detection using NetFlow/IPFIX, ACM SIGCOMM Computer Communication Review 44 (5) (2014) 20–26. doi:10.1145/2677046.2677050.

[68] E. K. Viegas, A. O. Santin, L. S. Oliveira, Toward a reliable anomaly-based intrusion detection in real-world environments, Computer Networks 127 (2017) 200–216. doi:10.1016/j.comnet.2017.08.013.

[69] P. Gogoi, M. H. Bhuyan, D. Bhattacharyya, J. K. Kalita, Packet and Flow Based Network Intrusion Dataset, in: International Conference on Contemporary Computing, Springer, 2012, pp. 322–334. doi:10.1007/978-3-642-32129-0_34.

[70] M. H. Bhuyan, D. K. Bhattacharyya, J. K. Kalita, Towards Generating Real-life Datasets for Network Intrusion Detection, International Journal of Network Security (IJNS) 17 (6) (2015) 683–701.

[71] A. Sperotto, R. Sadre, F. Van Vliet, A. Pras, A Labeled Data Set for Flow-Based Intrusion Detection, in: International Workshop on IP Operations and Management, Springer, 2009, pp. 39–50. doi:10.1007/978-3-642-04968-2_4.

[72] F. Gringoli, L. Salgarelli, M. Dusi, N. Cascarano, F. Risso, et al., GT: Picking up the Truth from the Ground for Internet Traffic, ACM SIGCOMM Computer Communication Review 39 (5) (2009) 12–18. doi:10.1145/1629607.1629610.

[73] M. J. Turcotte, A. D. Kent, C. Hash, Unified Host and Network Data Set, arXiv preprint arXiv:1708.07518.

[74] CYBER Systems and Technology, (Date last accessed 22-June-2018). URL https://www.ll.mit.edu/ideval/docs/index.html

[75] J. McHugh, Testing Intrusion Detection Systems: A Critique of the 1998 and 1999 DARPA Intrusion Detection System Evaluations As Performed by Lincoln Laboratory, ACM Transactions on Information and System Security (TISSEC) 3 (4) (2000) 262–294. doi:10.1145/382912.382923.

[76] G. Szabó, D. Orincsay, S. Malomsoky, I. Szabó, On the Validation of Traffic Classification Algorithms, in: International Conference on Passive and Active Network Measurement, Springer, 2008, pp. 72–81. doi:10.1007/978-3-540-79232-1_8.

[77] R. Fontugne, P. Borgnat, P. Abry, K. Fukuda, MAWILab: Combining Diverse Anomaly Detectors for Automated Anomaly Labeling and Performance Benchmarking, in: International Conference on emerging Networking EXperiments and Technologies (CoNEXT), ACM, New York, NY, USA, 2010, pp. 8:1–8:12. doi:10.1145/1921168.1921179.

[78] M. Hatada, M. Akiyama, T. Matsuki, T. Kasama, Empowering Anti-malware Research in Japan by Sharing the MWS Datasets, Journal of Information Processing 23 (5) (2015) 579–588. doi:10.2197/ipsjjip.23.579.

[79] D. Brauckhoff, A. Wagner, M. May, FLAME: A Flow-Level Anomaly Modeling Engine, in: Workshop on Cyber Security Experimentation and Test (CSET), USENIX Association, 2008, pp. 1:1–1:6.

[80] E. Vasilomanolakis, C. G. Cordero, N. Milanov, M. Mühlhäuser, Towards the creation of synthetic, yet realistic, intrusion detection datasets, in: IEEE Network Operations and Management Symposium (NOMS), IEEE, 2016, pp. 1209–1214. doi:10.1109/NOMS.2016.7502989.

[81] P. Siska, M. P. Stoecklin, A. Kind, T. Braun, A Flow Trace Generator using Graph-based Traffic Classification Techniques, in: International Wireless Communications and Mobile Computing Conference (IWCMC), ACM, 2010, pp. 457–462. doi:10.1145/1815396.1815503.

[82] M. Ring, D. Schlör, D. Landes, A. Hotho, Flow-based Network Traffic Generation using Generative Adversarial Networks, Computer & Security 82 (2019) 156–172. doi:10.1016/j.cose.2018.12.012.

[83] F. Erlacher, F. Dressler, How to Test an IDS?: GENESIDS: An Automated System for Generating Attack Traffic, in: Workshop on Traffic Measurements for Cybersecurity (WTMC), ACM, New York, NY, USA, 2018, pp. 46–51. doi:10.1145/3229598.3229601.

[84] S. Molnár, P. Megyesi, G. Szabo, How to Validate Traffic Generators?, in: IEEE International Conference on Communications Workshops (ICC), IEEE, 2013, pp. 1340–1344. doi:10.1109/ICCW.2013.6649445.

[85] G. Brogi, V. V. T. Tong, Sharing and replaying attack scenarios with Moirai, in: RESSI 2017: Rendez-vous de la Recherche et de l'Enseignement de la Sécurité des Systèmes d'Information, 2017.

[86] N. Rajasinghe, J. Samarabandu, X. Wang, INSecS-DCS: A Highly Customizable Network Intrusion Dataset Creation Framework, in: Canadian Conference on Electrical & Computer Engineering (CCECE), IEEE, 2018, pp. 1–4. doi:10.1109/CCECE.2018.8447661.

[87] F. J. Aparicio-Navarro, K. G. Kyriakopoulos, D. J. Parish, Automatic Dataset Labelling and Feature Selection for Intrusion Detection Systems, in: IEEE Military Communications Conference (MILCOM), IEEE, 2014, pp. 46–51. doi:10.1109/MILCOM.2014.17.

[88] R. Hofstede, A. Pras, A. Sperotto, G. D. Rodosek, Flow-Based Compromise Detection: Lessons Learned, IEEE Security Privacy 16 (1) (2018) 82–89. doi:10.1109/MSP.2018.1331021.

[89] J. Han, J. Pei, M. Kamber, Data Mining: Concepts and Techniques, 3rd Edition, Elsevier, 2011.

[90] M. Cermak, T. Jirsik, P. Velan, J. Komarkova, S. Spacek, M. Drasar, T. Plesnik, Towards Provable Network Traffic Measurement and Analysis via Semi-Labeled Trace Datasets, in: Network Traffic Measurement and Analysis Conference (TMA), IEEE, 2018, pp. 1–8. doi:10.23919/TMA.2018.8506498.

[91] G. Wang, J. Hao, J. Ma, L. Huang, A new approach to intrusion detection using Artificial Neural Networks and fuzzy clustering, Expert Systems with Applications 37 (9) (2010) 6225–6232. doi:10.1016/j.eswa.2010.02.102.

[92] J. Zhang, M. Zulkernine, A. Haque, Random-Forests-Based Network Intrusion Detection Systems, IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews) 38 (5) (2008) 649–659. doi:10.1109/TSMCC.2008.923876.

[93] M. V. Mahoney, Network Traffic Anomaly Detection based on Packet Bytes, in: ACM Symposium on Applied Computing, ACM, 2003, pp. 346–350. doi:10.1145/952532.952601.

[94] C. M. Inacio, B. Trammell, YAF: Yet Another Flowmeter, in: Large Installation System Administration Conference, 2010, pp. 107–118.

[95] J. Xu, J. Fan, M. H. Ammar, S. B. Moon, Prefix-Preserving IP Address Anonymization: Measurement-based security evaluation and a new cryptography-based Scheme, in: IEEE International Conference on Network Protocols, IEEE, 2002, pp. 280–289. doi:10.1109/ICNP.2002.1181415.

[96] R. Pang, M. Allman, V. Paxson, J. Lee, The Devil and Packet Trace Anonymization, ACM SIGCOMM Computer Communication Review 36 (1) (2006) 29–38. doi:10.1145/1111322.1111330.

[97] D. J. Kelly, R. A. Raines, M. R. Grimaila, R. O. Baldwin, B. E. Mullins, A Survey of State-of-the-Art in Anonymity Metrics, in: ACM Workshop on Network Data Anonymization, ACM, 2008, pp. 31–40. doi:10.1145/1456441.1456453.